Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) An audio/video filing system for handling and organizing audio/video data comprising:

a storage layer; and

an application programming interface residing in the audio/video filing system, said application programming interface having a first interface that controls transfer of information for handling audio/video data files and non-audio/video files at said storage layer, said application programming interface adapted to select a first set of function calls to manipulate said storage layer when ~~a~~ an audio/video file is detected and to select a second set of function calls to manipulate said storage layer when a non-audio/video file is detected.

2. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 1, further comprising:

a second interface which controls transfer of information between a controller capable of handling asynchronous data.

3. (Cancelled)

4. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 2, wherein said controller is capable of processing commands transmitted using protocol 61883.

5. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 4, wherein said audio/video file is transmitted using protocol

61833 in an isochronous manner and control components of the are transmitted in an asynchronous manner.

6. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 2, wherein said controller is a SBP-2 controller.

7. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 6, wherein said SBP-2 controller is capable of processing commands transmitted using serial-bus-protocol-2.

8. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 7, wherein said commands are transmitted using serial-bus-protocol-2 in an asynchronous manner.

9. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 1, wherein transfer of information to and from said audio/video filing system is independent of said storage layer implementation.

10. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 2, wherein control of said transfer of information to and from said controller is independent of internal implementation of said controller.

11. (Cancelled)

12. (Cancelled)

13. (Cancelled)

14. (Currently Amended)) The audio/video filing system ~~application programming interface~~ according to claim 1, wherein said audio/video data files is smaller than said non-audio/video files.

15. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 1, wherein said second set of function calls enable said audio/video file system to play or record a plurality of audio/video data files concurrently.

16. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 15, wherein said second set of function calls enable said audio/video file system to play or record said plurality of audio/video data files concurrently by using a channel ID parameter and an object ID parameter.

17. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 1, wherein said second set of function calls enable said audio/video file system to play and record an audio/video data stream concurrently.

18. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 17, wherein said second set of function calls enable said audio/video file system to play and record said audio/video data stream concurrently by using a channel ID parameter and an object ID parameter.

19. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 1, wherein one or more of said plurality of function calls are designed to allow said audio/video file system to initiate a play or record operation starting from within an audio/video file.

20. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 19, wherein said second set of function calls enable said

audio/video file system to initiate a play or record operation starting from within said audio/video file by using an offset parameter.

21. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 1, wherein said second set of function calls enable said audio/video file system to optimize disk access.

22. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 21, wherein said second set of function calls enable said audio/video file system to optimize disk access by designating a first group of function calls to handle a first type of file and a second group of function calls to handle a second type of file.

23. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 22, wherein said first type of file is a non-audio/video file; and wherein said second type of file is an audio/video file.

24. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 1, wherein said second set of function calls enable said audio/video file system to perform a plurality of trick operations with a data stream.

25. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 24, wherein said plurality of trick operations includes a plurality of forward operations.

26. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 25, wherein said plurality of forward operations includes a fast-forward operation, a slow- forward operation, and a step-forward operation.

27. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 24, wherein said plurality of trick operations includes a plurality of reverse operations.

28. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 27, wherein said plurality of reverse operations includes a fast-reverse operation, a slow- reverse operation, and a step-reverse operation.

29. (Currently Amended) An audio/video file system capable of handling and organizing audio/video data comprising:

an application programming interface residing in the audio/video filing system and adapted to interface with a plurality of controllers and to select a first plurality of function calls to manipulate said audio/video filing system when a first file type is detected and to select a second plurality of function calls to manipulate said audio/video filing system when a second file type is detected; said first plurality of function calls including:

a load function call designed to cause retrieval of descriptor information from a storage medium;

a store function call designed to cause storing of said descriptor information onto said storage medium;

a delete function call designed to cause deletion of said descriptor information from said storage medium; and

said second plurality of function calls including:

a play function call designed to cause a specified file to be played; a record function call designed to cause specified data to be recorded; and

a stop function call designed to cause a play or record operation to be stopped.

30. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 29, wherein said first plurality of function calls is designed to

handle a first type of file; and wherein said second plurality of function calls is designed to handle a second type of file.

31. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 30, wherein said first type of file is a non-audio/video file; and wherein said second type of file is an audio/video file.

32. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 29, wherein said first plurality of function calls further includes:

a validity function call designed to verify validity of a specified descriptor; and

wherein said second plurality of function calls further includes:

a pause function call designed to cause a play or record operation to be paused;

a resume function call designed to cause a previously paused operation to resume; and

an address retrieval function call designed to determine a logical block address of said specified file during a play or a record operation.

33. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 29, wherein said second plurality of function calls includes:

a plurality of function calls designed to cause forward operations to be performed; and

a plurality of function calls designed to cause reverse operations to be performed.

34. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 33, wherein said second plurality of function calls are designed to cause forward operations to be performed and include:

a fast-forward function call;

a slow-forward function call; and

a step-forward function call.

35. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 33, wherein said second plurality of function calls are designed to cause reverse operations to be performed include:

a fast-reverse function call;

a slow-reverse function call; and

a step-reverse function call.

36. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 29, wherein said application programming interface is capable of being used by a controller capable of handling isochronous and asynchronous data to communicate with said audio/video file system.

37. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 36, wherein said controller is an audio/video controller.

38. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 36, wherein said application programming interface enables a second controller capable of handling asynchronous data to communicate with said audio/video file system.

39. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 38, wherein said second controller is a SBP-2 controller.

40. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 32, wherein said specified descriptor is an object descriptor.

41. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 32, wherein said specified descriptor is a content list.

42. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 32, wherein said specified descriptor is a performance list.

43. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 32, wherein said specified descriptor is an EMS table.

44. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 32, wherein each of said first and second plurality of function calls is capable of passing a plurality of parameters.

45. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 44, wherein said plurality of parameters that is capable of being passed by said load function call includes a descriptor ID parameter, a type parameter, an offset parameter, a size parameter, a data_location parameter, and a call_back parameter.

46. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 44, wherein said plurality of parameters that is capable of being passed by said store function

call includes a descriptor ID parameter, a type parameter, an offset parameter, a size parameter, a data_location parameter, and a call_back parameter.

47. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 44, wherein said plurality of parameters that is capable of being passed by said delete function call includes a descriptor ID parameter, a type parameter, and a callback parameter.

48. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 44, wherein said plurality of parameters that is capable of being passed by said play function call includes a channel ID parameter, an object ID parameter, a start_position parameter, an_end position parameter, a speed parameter, and a call_back parameter.

49. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 44, wherein said plurality of parameters that are capable of being passed by said record function call include a channel ID parameter, an object ID parameter, a start_position parameter, a type parameter, and a call_back parameter.

50. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 44, wherein said plurality of parameters that is capable of being passed by said stop function call includes a channel ID parameter, a call_back parameter, and a logical_byte_address parameter.

51. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 44, wherein said plurality of parameters that is capable of being passed by said pause function call includes a channel ID parameter, a call_back parameter, and a logical_byte_address parameter.

52. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 44, wherein said plurality of parameters that is capable of being passed by said resume function call includes a channel ID parameter and a call_back parameter.

53. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 44, wherein said plurality of parameters that is capable of being

passed by said address retrieval function call includes a channel ID parameter and a count parameter.

54. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 44, wherein said plurality of parameters that is capable of being passed by said validity function call includes a descriptor ID parameter, a type parameter and a call_back parameter.

55. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 34, wherein said fast-forward function call is capable of passing a plurality of parameters including a channel ID parameter, a type parameter, an interval parameter, a repeat parameter, and a call_back parameter.

56. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 34, wherein said slow-forward function call is capable of passing a plurality of parameters including a channel ID parameter, a repeat parameter, an increment parameter and a callback parameter.

57. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 34, wherein said step-forward function call is capable of passing a plurality of parameters including a channel ID parameter, an increment parameter and a call_back parameter.

58. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 35, wherein said fast-reverse function call is capable of passing a plurality of parameters including a channel ID parameter, a type parameter, an interval parameter, a repeat parameter, and a call_back parameter.

59. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 35, wherein said slow-reverse function call is capable of passing a plurality of parameters including a channel ID parameter, a repeat parameter, an increment parameter and a call_back parameter.

60. (Currently Amended) The audio/video filing system ~~application programming interface~~ according to claim 35, wherein said step-reverse function call is capable of passing a plurality of parameters including a channel ID parameter, an increment parameter and a call_back parameter.

61. (Currently Amended) A method for providing communication with an audio/video file system, comprising steps of:

providing a first interface residing in the audio/video filing system, the first interface controlling~~which controls~~ transfers of information between said audio/video system and a first controller capable of handling isochronous and asynchronous data;

using a first plurality of function calls to manipulate said audio/video filing system when a descriptor file is detected by said first interface and a second plurality of function calls to manipulate said audio/video filing system when an audio/video file type is detected by said first interface; and

providing a second interface residing in the audio/video filing system, the first interface controlling~~which controls~~ transfers of information between said audio/video system and a second controller capable of handling asynchronous data.

62. (Previously presented) The method according to claim 61, wherein said signals transferred between said audio/video system and said first controller are independent of internal implementation of said first device; and wherein said signals transferred between said audio/video system and said second controller are independent of internal implementation of said second device.